



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/553,971	04/21/2000	Sai V. Allavaru	5181-48600	6569

7590 12/18/2003

Robert C Kowert
Conley Rose & Tayon P C
P O Box 398
Austin, TX 78767

EXAMINER

SHAH, NILESH R

ART UNIT	PAPER NUMBER
----------	--------------

2127

DATE MAILED: 12/18/2003

//

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/553,971

Applicant(s)

ALLAVARPU ET AL.

Examiner

Nilesh R Shah

Art Unit

2127

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 22 September 2003.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-38 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-38 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 21 April 2000 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. §§ 119 and 120

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.
- 13) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application) since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.
- a) ☐ The translation of the foreign language provisional application has been received.
- 14) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121 since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

Attachment(s)

- 1) ☐ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☒ Information Disclosure Statement(s) (PTO-1449) Paper No(s) 6,9.
- 4) ☐ Interview Summary (PTO-413) Paper No(s). _____.
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: _____.

DETAILED ACTION***Response to Amendment***

As per claim 1, Applicant states Drury does not teach the use of a primary and secondary scheduler. Drury clearly teaches the use of a primary and secondary scheduler (col. 8 line 1-col. 9 line 41) ('The embodiment of FIG. 1 may also be implemented with an enhanced scheduler 301. The enhanced scheduler 301 operates essentially identically to the scheduler 103 except the enhanced scheduler 301 incorporates a storage unit 302 and uses a more complex method to locate an appropriate binding. FIG. 3 shows the elements directly affected by the use of an enhanced scheduler 301.').

In addition applicant states Drury does not teach the use of requests from a multi-threaded application. Drury teaches the use of multi-threaded requests (col.2 line 64 –col.3 line 59, col. 8 lines 1-19) ('When an appropriate interface is found in the namespace 107, the scheduler 103 transfers to the client 101, a binding to a multi-threaded server 113. ') ('The schedulers mediate all remote procedure calls (RPCs) made by client processes to server processes using a scheduler and/or namespace accessible by the DCE processor. The scheduler database stores interfaces of single-thread servers, and the namespace stores interfaces of multi-thread servers. The scheduler, in response to receiving an identity of an interface from a client process searching the scheduler database and namespace to locate the interface. Upon locating the interface, the interface is provided to the client process so that client and server processes can be bound.'). The multi –threaded information is passed the scheduler thus the scheduler must be able to handle mutli-threaded information.

As per claim 15 and 27 Applicant states Drury does not teach the use of a primary and secondary scheduler. Drury clearly teaches the use of a primary and secondary scheduler (col. 8 line 1-col. 9 line 41) ('The embodiment of FIG. 1 may also be implemented with an enhanced scheduler 301. The enhanced scheduler 301 operates essentially identically to the scheduler 103 except the enhanced scheduler 301 incorporates a storage unit 302 and uses a more complex method to locate an appropriate binding. FIG. 3 shows the elements directly affected by the use of an enhanced scheduler 301.').

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

Art Unit: 2127

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1-7, 12-19, 23-31, 34-38 are rejected under 35 U.S.C. 102(b) as being anticipated by Drury et al (5,452,459) (hereinafter Drury).

As per claim 1 Drury teaches a thread-safe scheduler system comprising:

a primary scheduler which is executable to schedule requests for networked data resources (col. 8 lines 1-19) ('Sometimes, the scheduler 103 finds an appropriate yet unavailable interface in the scheduler DB 105. Since multi-process servers 114 are single threaded, the scheduler cannot assign multiple clients to a single server. Therefore, if no appropriate interfaces are available, the scheduler 103 places the client 101 in its queue.');

and
a secondary scheduler, wherein the secondary scheduler is executable to receive a plurality of requests from a multi-threaded application in a thread-safe manner and send the requests to the primary scheduler in a thread-safe manner (col. 8 lines 1-19) ('When an appropriate interface is found in the namespace 107, the scheduler 103 transfers to the client 101, a binding to a multi-threaded server 113. If no threads are available, the standard RPC queuing mechanism activates.')

As per claim 2 Drury teaches a system, wherein the primary scheduler is single-threaded (col. 8 lines 1-19) ('Sometimes, the scheduler 103 finds an appropriate yet unavailable interface in the scheduler DB 105. Since multi-process servers 114 are single threaded, the scheduler cannot assign multiple clients to a single server. Therefore, if no appropriate interfaces are available, the scheduler 103 places the client 101 in its queue.')

As per claim 3, Drury teaches a system, wherein the secondary scheduler is multi-threaded (col. 8 lines 1-19) ('When an appropriate interface is found in the namespace 107, the scheduler 103 transfers to the client 101, a binding to a multi-threaded server 113. If no threads are available, the standard RPC queuing mechanism activates.')

As per claim 4, Drury teaches a system wherein the secondary scheduler is executable to receive the plurality of requests from the multi-threaded application through a lock in a thread-safe manner (col. 3 line 60- col. 4 line 40) ('under this scheme, if a thread must execute non-reentrant code then it must first acquire the global lock. Once a thread has acquired a global lock it can execute to completion, secure in the knowledge that no other thread will execute simultaneously utilizing the resources controlled by the global lock.')

As per claim 5, Drury teaches a system wherein the primary scheduler is executable to receive the plurality of requests from the secondary scheduler through a lock in a thread-safe manner (col. 3 line 60- col. 4 line 40) ('under this scheme, if a thread must execute non-reentrant code then it must first acquire the global lock. Once a thread has acquired a global lock it can execute to completion, secure in the knowledge that no other thread will execute simultaneously utilizing the resources controlled by the global lock.')

As per claim 6, Drury teaches a system wherein the resources comprise a management information server, wherein the requests comprise management requests, and wherein the multi-

Art Unit: 2127

threaded application comprises a multi-threaded manager application (col. 6 lines 5-50) ('Via their pre-established agreement, the transaction manager notifies the scheduler that the transaction is over so the server may be released')

As per claim 7, Drury teaches a system further comprising: a management information server coupled to the primary scheduler through a management interface, wherein the primary scheduler is operable to send the requests to one or more managed objects through the management information server (col. 6 lines 5-50) ('Via their pre-established agreement, the transaction manager notifies the scheduler that the transaction is over so the server may be released')

As per claim 12, Drury teaches a system wherein the primary scheduler comprises a primary queue, which is operable to hold pending requests and responses to the requests (col. 8 lines 33-66) ('When acquiring a binding, the client 101 may notify the scheduler 103 and activate transactional context. As a result, the scheduler 103 alters its behavior.')

As per claim 13, Drury teaches a system wherein the secondary scheduler comprises a secondary queue which is operable to hold pending requests (col. 8 lines 33-66) ('When acquiring a binding, the client 101 may notify the scheduler 103 and activate transactional context. As a result, the scheduler 103 alters its behavior.')

As per claim 14, Drury teaches a system further comprising:

Art Unit: 2127

a communication pipe between the primary scheduler and secondary scheduler, wherein the secondary scheduler uses the communication pipe to wake the primary scheduler prior to sending one of the requests to the primary scheduler (col. 7 line 39 –col. 8 line 66) ('When an appropriate interface is found in the scheduler DB 105, the scheduler 103 transfers to the client 101 a binding for a multi-process server 114. The binding enables the client 101 to locate and access the desired server 114.')

As per claim 15, Drury teaches a thread-safe method for using a management interface for management of a plurality of managed objects on a network, the method comprising: receiving a plurality of management requests from a multi-threaded manager application into a secondary scheduler in a thread-safe manner (col.8 lines 1-66) ('When an appropriate interface is found in the namespace 107, the scheduler 103 transfers to the client 101, a binding to a multi-threaded server 113. If no threads are available, the standard RPC queuing mechanism activates.')

scheduling the plurality of management requests in a secondary queue in the secondary scheduler after receiving the management requests from the manager application (col.8 lines 1-66); sending the management requests from the secondary scheduler to a primary scheduler in a thread-safe manner (col.8 lines 1-66); scheduling the management requests in a primary queue in the primary scheduler (col.8 lines 1-66, col. 6 lines 5-50) ('Via their pre-established agreement, the transaction manager notifies the scheduler that the transaction is over so the server may be released'),

Art Unit: 2127

executing the management requests on the managed objects after scheduling the management requests in the primary queue (col.8 lines 1-66, col. 6 lines 5-50) ('Via their pre-established agreement, the transaction manager notifies the scheduler that the transaction is over so the server may be released')

As per claim 16 Drury teaches executing the management requests on the managed objects further comprises sending the management requests to a management information server coupled to the managed objects (col.8 lines 1-66) (col. 6 lines 5-50) ('Via their pre-established agreement, the transaction manager notifies the scheduler that the transaction is over so the server may be released')

As per claim 17 Drury teaches a method wherein each of the management requests comprises a corresponding callback function (col. 7 lines 38-68) ('First, the scheduler 103 calls the scheduler database 105 to check if any multi-process server 114 offers an appropriate interface.')

As per claim 18 Drury teaches a method further comprising:
receiving a response to one of the management requests from one of the managed objects after executing that management requests on one of the managed objects (col.8 lines 1-66) ('When an appropriate interface is found in the namespace 107, the scheduler 103 transfers to the client 101, a binding to a multi-threaded server 113. If no threads are available, the standard RPC queuing mechanism activates.');

executing the corresponding callback function for that management request to send the response to the multi-threaded manager application (col. 7 lines 38-68) ('First, the scheduler 103 calls the scheduler database 105 to check if any multi-process server 114 offers an appropriate interface.')

As per claim 19 Drury teaches a method further comprising:

enqueueing the response in the primary queue after receiving the response from one of the managed objects (col. 6 lines 5-50, col.8 lines 1-66, col. 7 lines 38-68) ('When an appropriate interface is found in the namespace 107, the scheduler 103 transfers to the client 101, a binding to a multi-threaded server 113. If no threads are available, the standard RPC queuing mechanism activates.')

finding the callback function corresponding to the response after enqueueing the response (col. 7 lines 38-68) ('First, the scheduler 103 calls the scheduler database 105 to check if any multi-process server 114 offers an appropriate interface.');

and
dequeueing the response from the primary queue before executing the corresponding callback function to send the response to the multi-threaded manager application (col. 7 lines 38-68) ('First, the scheduler 103 calls the scheduler database 105 to check if any multi-process server 114 offers an appropriate interface.').

As per claim 20, as applicant has already cited in disclosure on page 5 lines 10 –22. PMI is a single thread program. Drury teaches that a single thread can carry out a plurality of

Art Unit: 2127

functions which are operable to carry out the requests (col. 3 line 60- col. 4 line 40, col. 6 lines 5-50 and col. 8 lines 1-19).

As per claim 23 Drury teaches a method wherein the receiving the plurality of management requests from the multi-threaded manager application into the secondary scheduler in the thread-safe manner comprises receiving the plurality of management requests through a thread-safe lock (col. 3 line 60- col. 4 line 40) ('under this scheme, if a thread must execute non-reentrant code then it must first acquire the global lock. Once a thread has acquired a global lock it can execute to completion, secure in the knowledge that no other thread will execute simultaneously utilizing the resources controlled by the global lock.')

As per claim 24 Drury teaches a method, wherein sending the management requests to the primary scheduler in the thread-safe manner comprises dispatching the management requests through a thread-safe lock (col. 3 line 60- col. 4 line 40) ('under this scheme, if a thread must execute non-reentrant code then it must first acquire the global lock. Once a thread has acquired a global lock it can execute to completion, secure in the knowledge that no other thread will execute simultaneously utilizing the resources controlled by the global lock.')

As per claim 25 Drury teaches a method wherein the primary scheduler is executed in a single thread associated with the management interface, and wherein the secondary scheduler is executed in at least one different thread (col. 8 lines 1-19) ('Sometimes, the scheduler 103 finds an appropriate yet unavailable interface in the scheduler DB 105. Since multi-process servers

Art Unit: 2127

114 are single threaded, the scheduler cannot assign multiple clients to a single server. Therefore, if no appropriate interfaces are available, the scheduler 103 places the client 101 in its queue.')

('When an appropriate interface is found in the namespace 107, the scheduler 103 transfers to the client 101, a binding to a multi-threaded server 113. If no threads are available, the standard RPC queuing mechanism activates.').

As per claim 26 Drury teaches a method, wherein the secondary scheduler is multi-threaded (col. 8 lines 1-19) ('When an appropriate interface is found in the namespace 107, the scheduler 103 transfers to the client 101, a binding to a multi-threaded server 113. If no threads are available, the standard RPC queuing mechanism activates.').

Claim 27-32 35- 38 are rejected based on the same rejections as claims 15- 20 23-26 respectfully.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 8-9, 21-22 and 33-34 are rejected under 35 U.S.C. 103(a) as being unpatentable over Drury in view of Neufeld (5,974,438).

As per claim 8, Drury does not specifically teach the use of a telephone network. Neufeld teaches a system wherein the managed objects comprise one or more objects corresponding to a telephone network (col. 11 line 53 –col. 12 line 11). ('The multi-processor computer system is generally indicated by the numeral 100 and comprises processors 102a-d, first level caches 104a-d, second level caches 106a-d, third level caches 107a-d, main memory 108, PCI/memory bridges and cache controllers 110a-d, SCSI adapter 112, LAN adapter 114, ISA adapter 116, user interface 118, video adapter 120, video monitor 122, PCI bus 124, hard disk 126, tape drive 128, CD ROM drive 130, local area network 132, floppy disk 134, telephone modem 136, keyboard 140 and a cursor pointing device'). It would have been obvious to one skilled in the art to add the teachings of Neufeld to Drury in order to have schedulers in different locations, which are able to communicate with each other over a telephone network.

As per claim 9, Drury does not specifically teach the use of a telephone network. Neufeld teaches a system wherein the managed objects comprise an object to corresponding to a telecommunications device (col. 11 line 53 –col. 12 line 11). ('The multi-processor computer system is generally indicated by the numeral 100 and comprises processors 102a-d, first level caches 104a-d, second level caches 106a-d, third level caches 107a-d, main memory 108, PCI/memory bridges and cache controllers 110a-d, SCSI adapter 112, LAN adapter 114, ISA adapter 116, user interface 118, video adapter 120, video monitor 122, PCI bus 124, hard disk 126, tape drive 128, CD ROM drive 130, local area network 132, floppy disk 134, telephone

Art Unit: 2127

modem 136, keyboard 140 and a cursor pointing device'). It would have been obvious to one skilled in the art to add the teachings of Neufeld to Drury in order to have schedulers in different locations, which are able to communicate with each other over a telephone network.

As per claim 21, Drury does not specifically teach the use of a telephone network. Neufeld teaches a system wherein the managed objects comprise one or more objects corresponding to a telephone network (col. 11 line 53 –col. 12 line 11). ('The multi-processor computer system is generally indicated by the numeral 100 and comprises processors 102a-d, first level caches 104a-d, second level caches 106a-d, third level caches 107a-d, main memory 108, PCI/memory bridges and cache controllers 110a-d, SCSI adapter 112, LAN adapter 114, ISA adapter 116, user interface 118, video adapter 120, video monitor 122, PCI bus 124, hard disk 126, tape drive 128, CD ROM drive 130, local area network 132, floppy disk 134, telephone modem 136, keyboard 140 and a cursor pointing device'). It would have been obvious to one skilled in the art to add the teachings of Neufeld to Drury in order to have schedulers in different locations, which are able to communicate with each other over a telephone network.

As per claim 22, Drury does not specifically teach the use of a telephone network.

Art Unit: 2127

Neufeld teaches a system wherein the managed objects comprise an object to corresponding to a telecommunications device (col. 11 line 53 –col. 12 line 11). ('The multi-processor computer system is generally indicated by the numeral 100 and comprises processors 102a-d, first level caches 104a-d, second level caches 106a-d, third level caches 107a-d, main memory 108, PCI/memory bridges and cache controllers 110a-d, SCSI adapter 112, LAN adapter 114, ISA adapter 116, user interface 118, video adapter 120, video monitor 122, PCI bus 124, hard disk 126, tape drive 128, CD ROM drive 130, local area network 132, floppy disk 134, telephone modem 136, keyboard 140 and a cursor pointing device'). It would have been obvious to one skilled in the art to add the teachings of Neufeld to Drury in order to have schedulers in different locations, which are able to communicate with each other over a telephone network.

Claims 33-34 are rejected based on the same rejections as claims 21-22 respectfully.

Claims 10-11, 20 and 32 are rejected under 35 U.S.C. 103(a) as being unpatentable over Drury.

As per claim 10, as applicant has already cited in disclosure on page 5 lines 10 –22. PMI is a single thread program and PMI is as event based API. Drury teaches that a single thread can carry out a plurality of functions, which are operable to carry out the requests as an API program. (col. 3 line 60- col. 4 line 40, col. 6 lines 5-50 and col. 8 lines 1-19). It would have been obvious to one skilled in the art to have PMI as event based API system as disclosed in the specification.

Art Unit: 2127

As per claim 11, Drury teaches wherein the requests comprise callback functions, and wherein the callback functions are executable to send responses to the requests to the multi-threaded application (col. 7 lines 38-68) ('First, the scheduler 103 calls the scheduler database 105 to check if any multi-process server 114 offers an appropriate interface.')

Claim 20 and 32 are rejected based on the same rejections as claim 10.

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Nilesh R Shah whose telephone number is 703-305-8105. The examiner can normally be reached on Monday-Friday 8am-4pm.

Art Unit: 2127

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, William Grant can be reached on 703-308-1108. The fax phone numbers for the organization where this application or proceeding is assigned are (703)305-0040 for regular communications.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is 703-305-3900.

NS

 **BANAN, MICHAEL A.**
PRIMARY EXAMINER